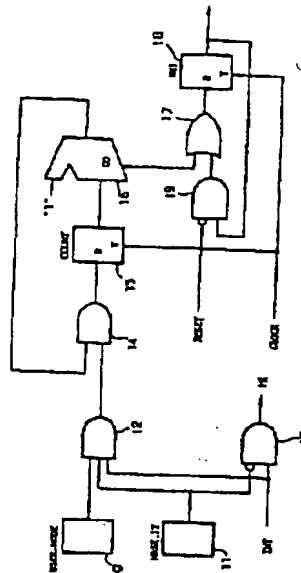


PUBLICATION NUMBER : JP4135246  
 PUBLICATION DATE : 08-05-92  
 ABSTRACT PUBLICATION DATE: 26-08-92  
 ABSTRACT VOLUME : 016404  
 APPLICATION DATE : 20-08-90  
 APPLICATION NUMBER : JP900218665  
 GROUP : P1410  
 APPLICANT : MITSUBISHI ELECTRIC CORP  
 INVENTOR : SUZUKI HISAAKI  
 INT.CL. : G06F9/46  
 TITLE : DATA PROCESSOR



**ABSTRACT :** PURPOSE: To prevent the occurrence of a system down as well as the adverse influence caused by the defective control of an interruption masks of a user program by counting a period during which the interruption mask is produced in a user mode and producing forcibly a mask unable interruption at detection of a time-out state.  
 CONSTITUTION: In a user mode, a USER-MODE latch 10 is set and at the same time the execution of a user program is started. Then an interruption mask time-out state detection means and the mask unable interruption generation means 10-19 detect such a system state where an interruption masks state lasts for a long period in the user mode and at the same time the mask enable interruption is not received for a fixed period after generation of the interruption factor. Then the execution of the user program that caused system state is interrupted and at the same time a masked masks enable interruption can be accepted. As a result, the reception of the interruption mask is continuously suppressed and therefore the system operation is never adversely affected.

平4-135245 (6)

A  
B  
C  
D  
E 1 , vt 2  
E 1 , vt 3  
E 1 , vt 4  
E 5 , vt 6  
E 7 , vt 8

2 , vt 3  
E 1 , vt 5  
E 1 , vt 6

⑯ 日本国特許庁 (JP)

⑯ 特許出願公開

## ⑯ 公開特許公報 (A)

平4-135246

⑯ Int. CL<sup>5</sup>

G 06 F 9/46

識別記号

3 1 1 E

庁内整理番号

8120-5B

⑯ 公開 平成4年(1992)5月8日

審査請求 未請求 請求項の数 1 (全6頁)

⑯ 発明の名称 データ処理装置

⑯ 特 願 平2-218665

⑯ 出 願 平2(1990)8月20日

⑯ 発明者 鈴木 寿明

神奈川県鎌倉市上町屋325番地 三菱電機株式会社コンピュータ製作所内

⑯ 出願人 三菱電機株式会社

東京都千代田区丸の内2丁目2番3号

⑯ 代理人 弁理士 會我 道照

外6名

図の説明図

## 明細書

## 1. 発明の名称

データ処理装置

## 2. 特許請求の範囲

スーパバイザモードとユーザモードのどちらかのプログラム実行状態の設定手段と、該ユーザモードのプログラム実行状態にマスク可能割込みをマスクする割込みマスク手段と、該割込みマスク手段によってマスク可能割込みがマスクされている期間で、かつマスク可能割込み要因が発生してからの経過時間が予め定められた時間長を超えたか否かを検出する割込みマスクタイムアウト検出手段と、該割込みマスクタイムアウト検出手段からのタイムアウト通知に基づいて該マスク可能割込みとは異なる種類のマスク不可能割込みを発生させるマスク不可能割込み発生手段とを備えたデータ処理装置。

## 3. 発明の詳細な説明

## (産業上の利用分野)

この発明は、データ処理装置に関するもので、

特にコンピュータ分野における割込み処理方式に関するものである。

## (従来の技術)

第4図はユーザモード中に割込みマスクをセットして実行されるユーザプログラムの起動を示すフローチャートで、図において、(1) はスーパバイザモードにおいてユーザプログラムの実行を起動するディスパッチャ手続き、(2) と(4) はユーザプログラムの実行処理、(3) はユーザプログラム中でマスク可能な割込みのマスクをセットする手続きを示す。

次に動作について説明する。スーパバイザは実行すべきユーザプログラムを選択し、ユーザプログラムの実行を起動するディスパッチャ手続き(1) の処理を行う。

ディスパッチャ手続き(1) でユーザプログラムの起動をする際には、ユーザプログラムの誤動作等による影響の波及を最小限に止めるために、システム制御等の命令を特権命令として分類し、これらの特権命令の実行を許可しないユーザモード

に実行状態を切り換える。その後、ユーザモードでユーザプログラムの実行(1)が開始される。

ユーザプログラムの中には、自プログラム中のある特定の処理範囲内で外部からの影響を受けずに連続的に実行を行うために、マスク可能な割込みをマスクすることがある。通常では割込みマスクセッタ(3)を行い、割込み抑制状態を前提とした処理範囲のプログラム実行を終了したならば、即ち割込みマスクを解除し、その後残りの処理を実行するようにユーザプログラムは作られなければならない。しかし、プログラムの論理誤り等何等かの原因により割込みマスクの解除をせずに割込みマスクをセットしたまま、ユーザプログラムが実行される場合には、割込み要因が発生したとしても割込みマスクが解除されない限り、割込みが受けられずにユーザプログラムの実行が続行(4)されることになる。

(発明が解決しようとする課題)

従来のデータ処理装置は、以上のように、ユーザプログラムが割込みマスクをセットして走行し

ている時には、ユーザプログラムの挙動に割込みが制御が委ねられることになる。即ち、ユーザプログラムの論理誤りによる割込みマスクの解除がそれや一定時間を越えて長期間割込みのマスクの状態が続いた時等に、緊急度の高い割込み要因が発生しても割込みマスクによって受付が抑止され受け、システム動作に重大な悪影響を及ぼす可能性があった。

また、特開昭60-201438号及び特開昭60-108935号公報、特に上記特開昭60-201438号公報には、必要な時間に限って割込みを禁止し、かつ指定の時間経過後自動的に割込み可能に復す手段を持つ技術が開示されているが、この先行技術においては、割込み受付の可否を制御するマスク手段の機能に関して、マイクロプログラムがマスク有効期間(時間的仕様)を予め知った上でマスク有効化制御を主体的に且つ明示的に行う必要があり、また、タイマによって一定期間後に入る割込みの種類が、同一種類の割込みがマスク解除の形で入るようになっており、マスクの解除は行わず

に、別の種類の割込みを強制的に発生する形になっている。

この発明は上記の点を鑑みなされたもので、ユーザプログラムが一定期間以上割込みマスクをセットし続けた場合に割込み要因が出現してから一定時間が経過したことを検出し、マスクの解除は行わずにマスク不可能割込みを発生して最終的に解除できるデータ処理装置を得ることを目的としている。

(課題を解決するための手段)

この発明に係るデータ処理装置は、スーパーバイザモードとユーザモードのどちらかのプログラム実行状態の設定手段と、該ユーザモードのプログラム実行状態にマスク可能な割込みをマスクする割込みマスク手段と、該割込みマスク手段によってマスク可能な割込みがマスクされている期間で、かつマスク可能な割込み要因が発生してからの経過時間が予め定められた時間長を超えたか否かを検出する割込みマスクタイムアウト検出手段と、該割込みマスクタイムアウト検出手段からのタイムア

ウト通知に基づいて該マスク可能な割込みとは異なる種類のマスク不可能割込みを発生させるマスク不可能割込み発生手段とを備えたものである。

(作用)

この発明において、割込みマスクタイムアウト検出手段とマスク不可能割込み発生手段は、ユーザモードでの割込みマスク状態が長期間継続した時で、かつマスク可能な割込みがその要因発生から一定期間受けられないでいるようなシステム状態を検出し、このシステム状態を作り出した原因となったユーザプログラムの実行を中断させる手段を提供すると共に、マスクされていたマスク可能な割込みの受付を可能にする。

(実施例)

以下、この発明の一実施例を図について説明する。第1図は割込みマスクタイムアウト検出手段及びマスク不可能割込み発生手段の一実施例を示す論理図で、第1図において、(10)はユーザモードのプログラム実行状態時にセットされるUSER-MODEラッチ、(11)は割込みマスク時にセットされ

特開平4-135246 (3)

(2)

る MASK IT ラッチ、 (12) ないし (14) 及び (15) は論理 AND ゲート、 (16) は割込みマスク時にマスク可能割込み要因 (以下 INT と称す) が発生した後、 タイムアウトか否かをカウントする中間結果を保持する COUNT ラッチ、 (18) は COUNT ラッチ (15) の内容に 1 を加算し、その結果と最上位ビットからのキャリー (以下 CO と称す) を出力する 2 進加算器、 (17) は論理 OR ゲート、 (18) はマスク不可能割込み要求を保持する NMI ラッチである。

また、第 2 図は第 1 図中に示された論理の動作を説明するタイミングチャートで、図中の名称は第 1 図中に示されたものと同一箇所、同一意味である。

さうに、第 3 図はこの発明におけるユーザモード中に割込みマスクをセットして実行されるユーザプログラム及びスーパバイザモードで実行されるプログラムの挙動を示すフローチャートであり、 (5) はマスク不可能割込みによって起動される割込みハンドラ処理、 (6) は割込みマスクタイムアウトによる割込みかどうかを判定する手続

き、 (7) は割込みマスクタイムアウト要因となつたユーザプログラムのアボード処理手続きである。

次にこの発明の一実施例に基づいて動作を説明する。先ず、通常動作時について述べる。時刻  $t_a$  (第 2 図) でユーザモードに入ると、USER-MODE ラッチ (10) がセットされると共に、ユーザプログラムの実行が開始される。この時には USER-MODE ラッチ (10) はセットされているが、割込みマスク状態を示す MASK IT ラッチ (11) は未だセットされておらず、論理 AND ゲート (12) はゼロを出力する。すると、論理 AND ゲート (14) も閉じられて、 COUNT ラッチ (15) にはゼロの値がセットされる。今、COUNT ラッチ (15) のビット幅及び加算器 (16) の演算幅を各  $n$  ビット ( $n \geq 1$  でよいが、第 2 図では  $n \geq 2$  の例である) とし、加算器 (16) のキャリー CO が output される条件を以てタイムアウト検出と定義する。COUNT ラッチ (15) の出力がゼロである時には、加算器 (16) の出力、 CO は各々 1, 0 となり、タイムアウト検出はされず論理 OR ゲ

ト (17) を通してマスク不可能割込み要求を保持する NMI ラッチ (18) へのセットも行われない。

時刻  $t_b$  においてユーザプログラム中で割込みマスクをセットすると、 MASK IT ラッチ (11) がセットされる。しかし、マスク可能割込み要求 INT はこの時点では発生しておらず、論理 AND ゲート (12) は依然として 0 を出力し続けるため、論理 AND ゲート (14) が閉じられて、 COUNT ラッチ (15) には 0 がセットされ続ける。

次に、時刻  $t_c$  において、マスク可能割込み要求 INT がアサートされると、 MASK IT ラッチ (11) がセットされていることから、論理 AND ゲート (14) が閉じることによってマスク可能割込み要求 INT は受け付けられず、マスク可能割込み受付を示す M1 値は 0 となる。一方、マスク可能割込み要求 INT は論理 AND ゲート (12) に入力される。この時、既に USER-MODE ラッチ (10) と MASK IT ラッチ (11) が共にセットされていることから、論理 AND ゲート (12) は開き、 1 を出力する。論理 AND ゲート (12) の出力 1 は、論理 AND ゲート (14) に入力さ

れで加算器 (16) の出力を COUNT ラッチ (15) にセットする様に働く。COUNT ラッチ (15) の内容は、論理 AND ゲート (14) が閉じている間はデータ処理装置内のクロックである CLOCK が入っている毎にゼロがセットされていたので、加算器 (16) は  $0 + 1$  即ち 1 を出力し、時刻  $t_c$  以降で最初の CLOCK が発生時に COUNT ラッチ (15) には 1 がセットされる。そして、USER-MODE ラッチ (10)、 MASK IT ラッチ (11)、そしてマスク可能割込み INT の全てがセットされ続ける限り、 CLOCK に同期して COUNT ラッチ (15) の内容はプラス 1 ずつ増加していく。

時刻  $t_d$  において、割込みマスクが解除されて、 MASK IT ラッチ (11) がリセットされた場合には、論理 AND ゲート (12) が閉ざされることによって、論理 AND ゲート (14) も閉ざされてゼロを出力する。即ち、時刻  $t_d$  において、  $2^{\alpha} - K$  の値 (但し  $K > 2^{\alpha} - 1$ ) までカウントアップされた COUNT ラッチ (15) は、時刻  $t_d$  以降で最初の CLOCK 入力によってゼロにリセットされることになる。

一方、時刻  $t_d$  に MASK IT ラッチ (11) がリセット

特開平4-135246 (4)

されたことで、論理ANDゲート(13)は開かれ、マスク可能割込み要求INTは受けられてもNMI信号に1を出力する。この後、本データ処理装置は、割込み受付処理を開始し、時刻t<sub>0</sub>において、マスク可能割込み要求をリセットし、時刻t<sub>1</sub>で割込み処理ハンドラの実行を開始するためにユーザモードからスーパバイザモードにプログラム実行状態を切り換える。

次いで、この発明の要点である割込みマスクタイムアウト動作について説明する。

時刻t<sub>1</sub>(第3図)において、ユーザモードに入り、USER MODEラッチ(10)がセットされる。この時の動作は前記の時刻t<sub>0</sub>(第2図)と同様である。次に、時刻t<sub>2</sub>において、ユーザプログラムが割込みマスクをセットすることで、MASK ITラッチ(11)がセットされる。この時の動作も前記の時刻t<sub>0</sub>と同様になり、COUNTラッチ(15)には依然として0がセットされ続け、加算器(16)のCO出力にも0であり続ける。ユーザプログラムが割込みをマスクして動作している間に、時刻t<sub>3</sub>において、

マスク可能割込み要求INTが発生する。この時の動作も前述の時刻t<sub>0</sub>と同様となり、t<sub>3</sub>以後のCLOCKが入力される毎に、COUNTラッチ(15)の値は1-2-…とプラス1されていくことになる。

ここで、もし、ユーザプログラムが割込みマスク状態を解除せずに動作し続けると、COUNTラッチ(15)の値はCLOCKに同期してプラス1され続け、ついに2<sup>n</sup>-1の値に至る。COUNTラッチ(15)が2<sup>n</sup>-1の値になった時の加算器(16)では(2<sup>n</sup>-1)+1の加算を行うことになるが、演算幅はnビットであるので、2<sup>n</sup>の値を出力しようとすると、COは1、出力は0となる。つまり予め定義した様に、タイムアウト検出条件が成立することになる。この時出力されるCOは、論理ORゲート(17)を通過して、次にCLOCKが入力されるタイミングで、即ち、時刻t<sub>4</sub>でマスク不可能割込み要求を保持するNMIラッチ(18)をセットする。

一方、時刻t<sub>4</sub>では、COUNTラッチ(15)には2<sup>n</sup>の値の下位nビットのみをセットすることから前のサイクルの値2<sup>n</sup>-1から値0にタップアラウ

ンドすることになる。

以上のようにして、ユーザモードでの割込みマスクやタイムアウト条件を検出し、NMIラッチ(18)をセットするが、NMIラッチ(18)にセットされたマスク不可能割込み要求が本データ処理内で受け付けられる迄の間は、論理ANDゲート(13)及び論理ORゲート(17)によってホールドされる。マスク不可能割込み要求が受けられると、マスク不可能割込み要求をリセットする際には、RESET信号が入力され、論理ANDゲート(13)が閉じることでNMIラッチ(18)に0をセットする。この後、ユーザモードからスーパバイザモードにプログラム実行状態を切り換えて、割込みハンドラ処理(5)を開始する。この割込みハンドラ処理(5)において、割込み要因が割込みタイムアウトによるものかどうかを判定する手続き(6)により割込みマスクタイムアウトと判明すると、次いで原因となつたユーザプログラムのアボード処理手続き(7)を実行する。

なお、上記実施例では、タイムアウトの時間間

隔を(CLOCKの周期)×2<sup>n</sup>の形で定義しているが、nビット幅のアップカウンタを以ってタイムアウト検出を行うものを示したが、タイムアウト時間間隔をレジスタ等の保持手段を以って構成し、ダウンカウンタを以って構成する形でも良い。

#### (発明の効果)

以上のように、この発明によれば、ユーザモード時の割込みマスク(マスク可能割込み要因)が発生している期間を計測し、タイムアウト検出時にはマスク不可能割込みを強制的に発生させるように構成したので、ユーザプログラムの割込みマスク制御不良による悪影響やシステムダウンを防止でき、システム稼動率を高めることができると共に、異常状態の早期検出及び自動修復による可用性の向上を図ることができる。

#### 4. 図面の簡単な説明

第1図はこの発明による割込みマスクタイムアウト検出手段及びマスク不可能割込み発生手段の一実施例を示す論理図、第2図は第1図で示された論理の動作を説明するタイミングチャート、第

(4)

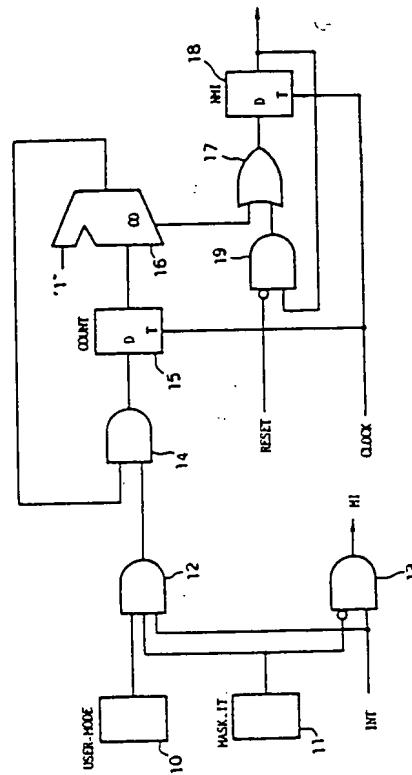
3図はこの発明におけるユーザモード中で割込みマスクをセットして実行されるユーザプログラム及びスーパバイザモードで実行されるプログラムの動作を示すフローチャート、第4図は従来のユーザモード中で割込みマスクをセットして実行されるユーザプログラム及びスーパバイザモードで実行されるプログラムの動作を示すフローチャートである。

(1) ...ディスパッチャ手続き、(2) ...ユーザプログラムの実行、(3) ...割込みマスクセット、(5) ...割込みハンドラ、(6) ...割込みマスクタイムアウト判定処理、(7) ...ユーザプログラムアボード処理、(10)~(19) ...割込みマスクタイムアウト検出手段とマスク不可能割込み発生手段の論理構成の各要素である。

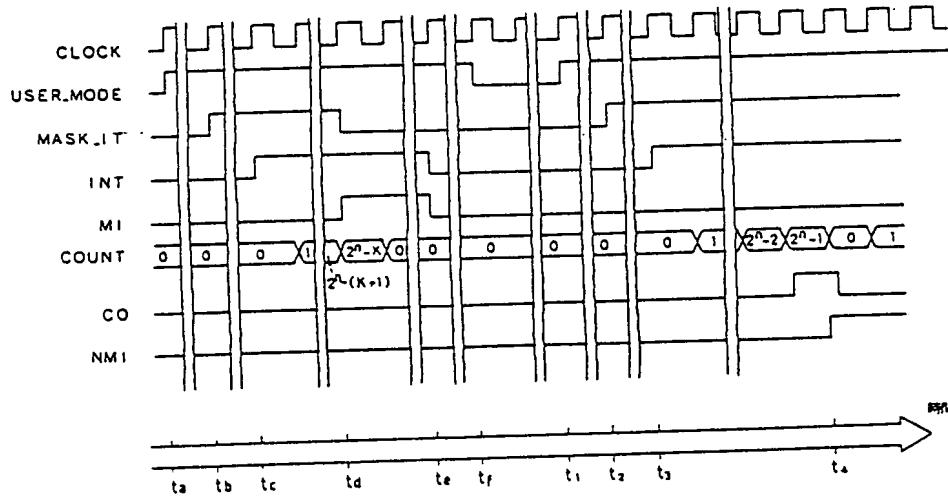
尚、図中、同一符号は同一又は相当部分を示す。

代理人 山崎宗秋

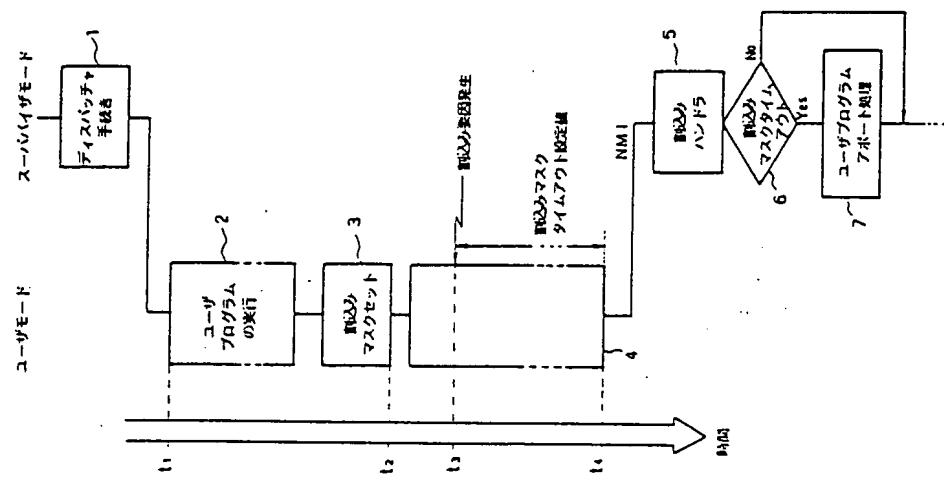
図 1 第



第 2 図



四三



第 4 四

